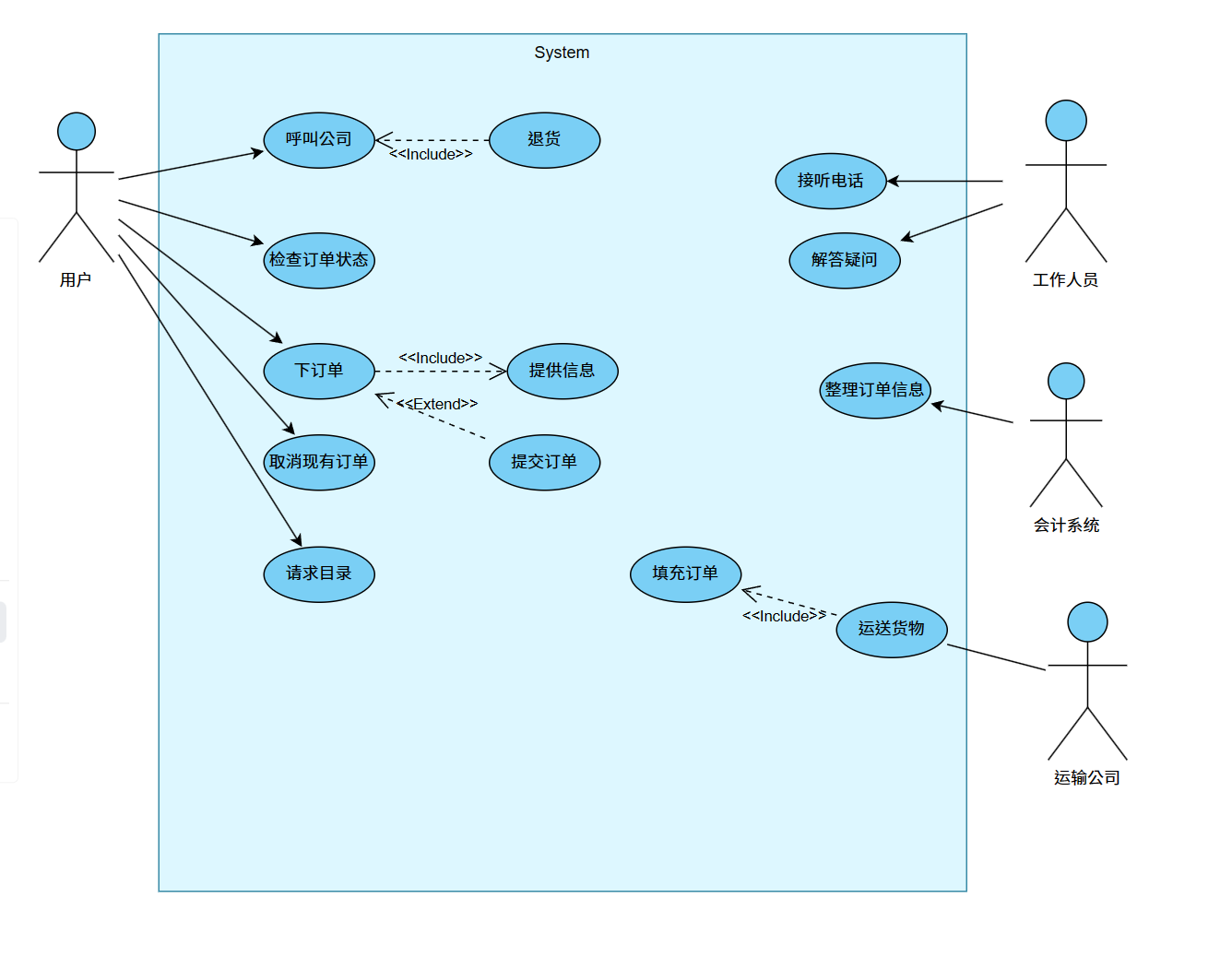
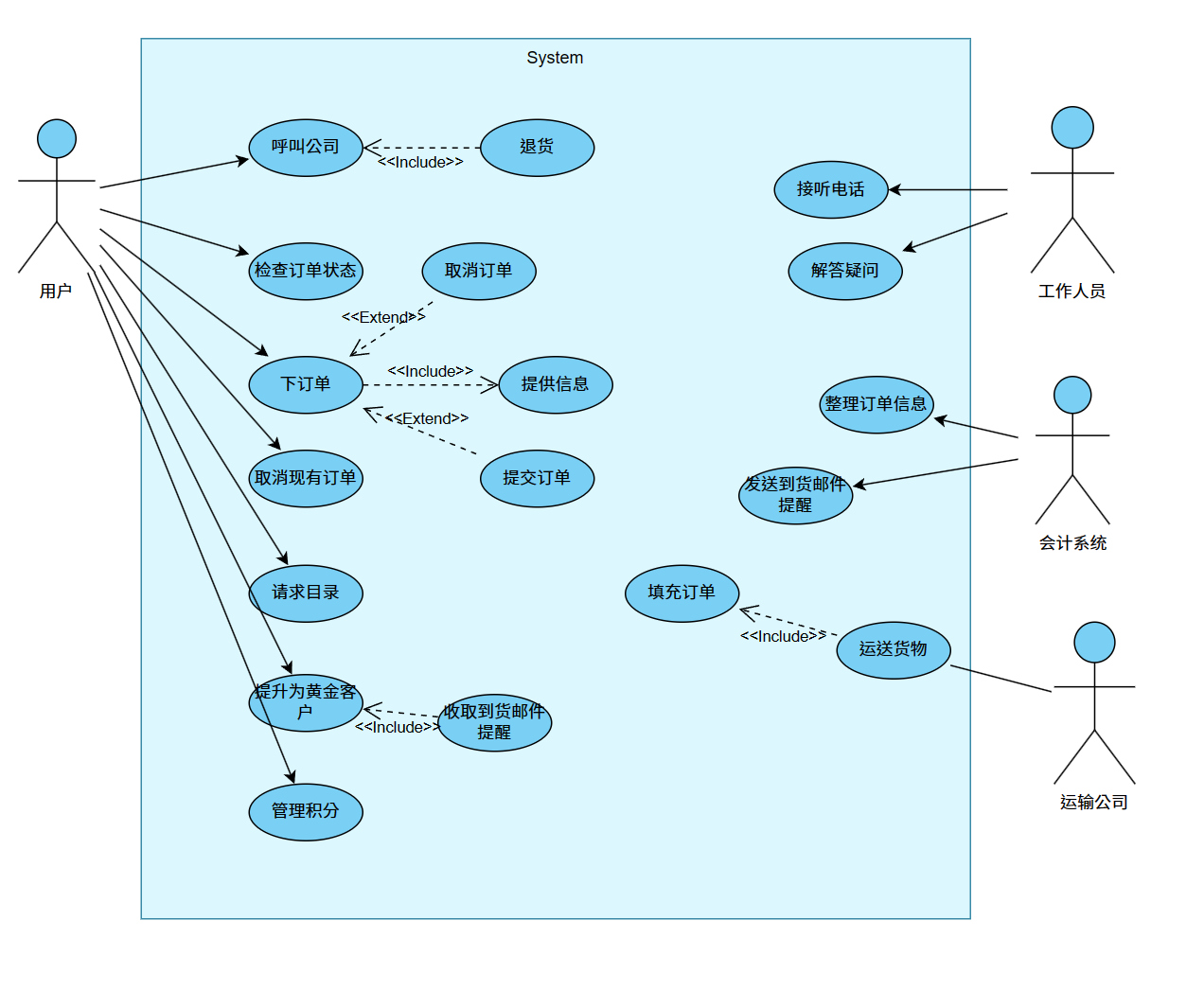
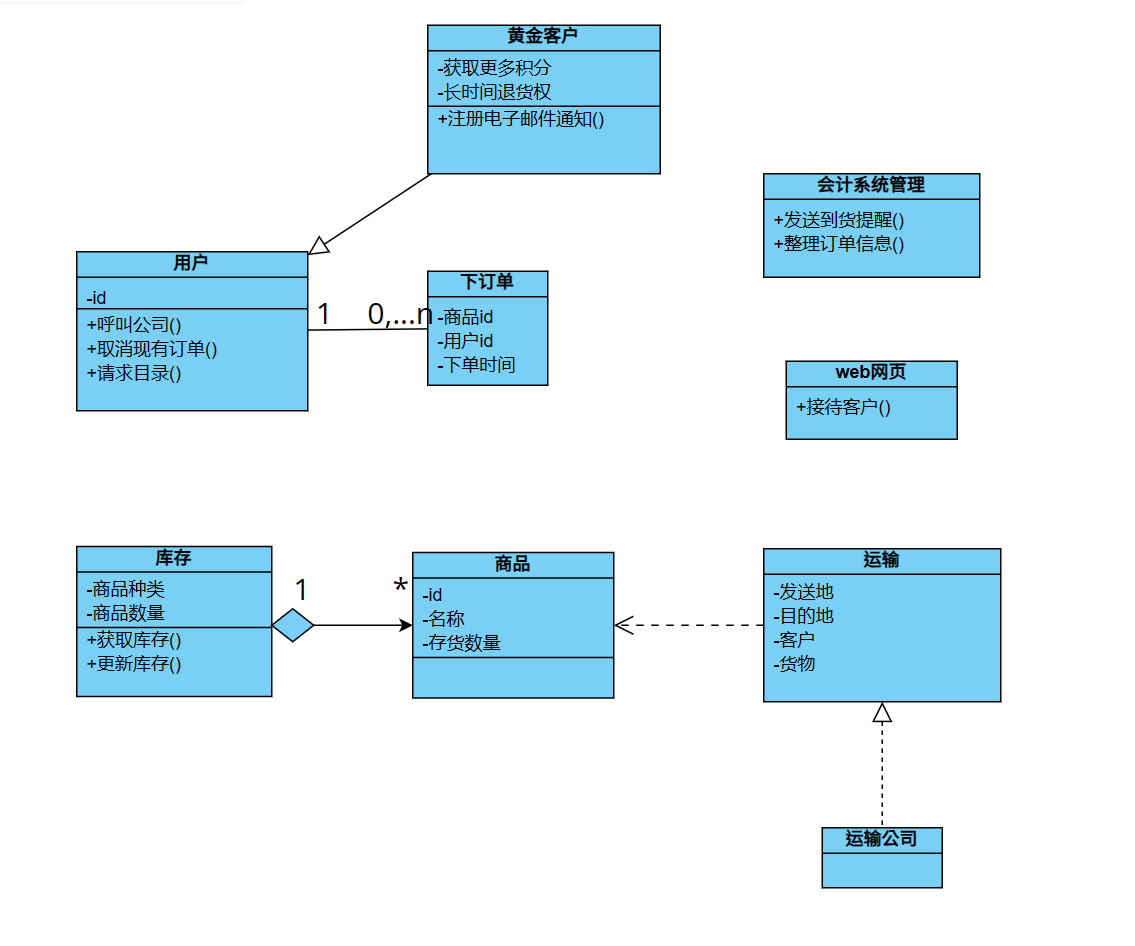
1. ai修改前：（花费时间：30min）



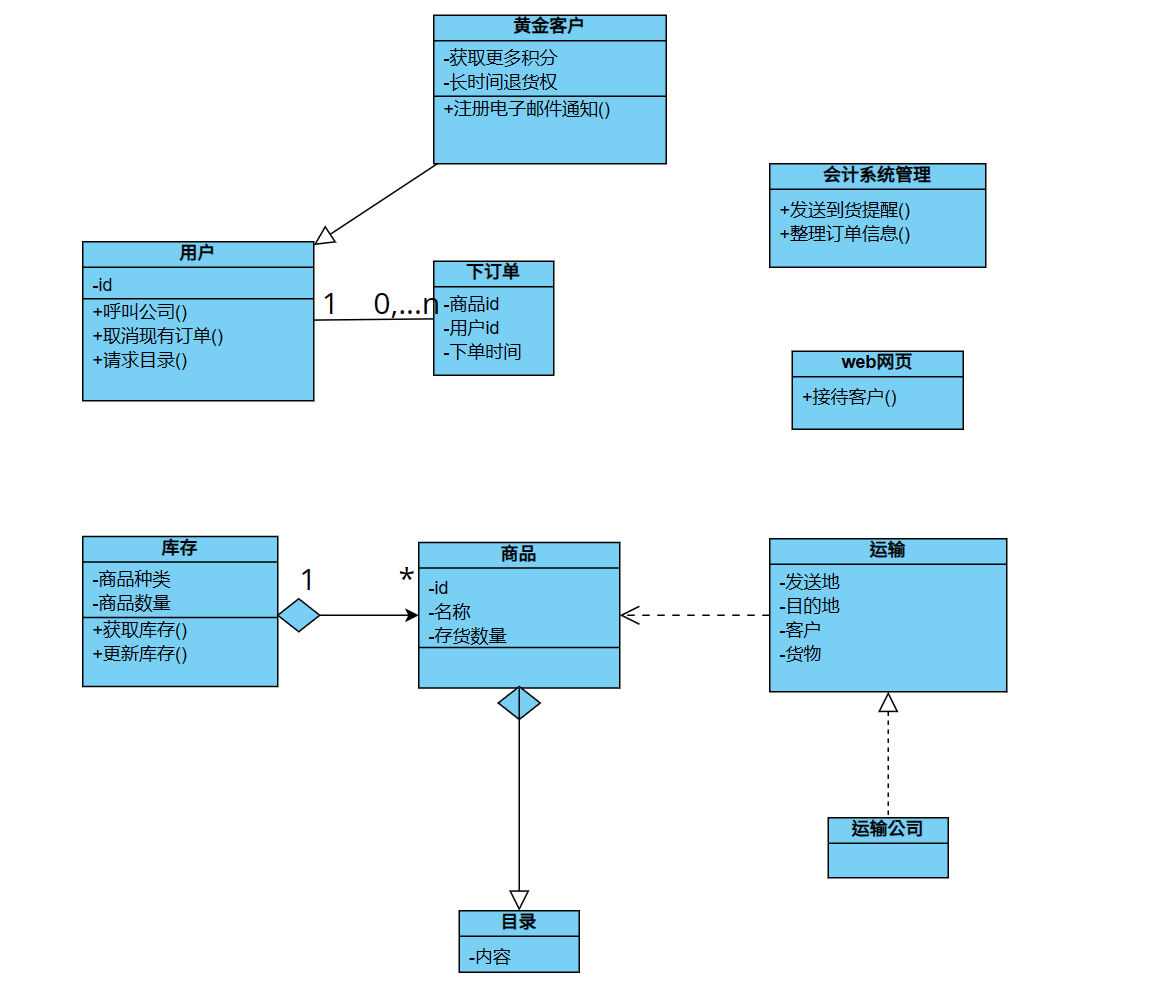
修改后：（花费时间：10min）



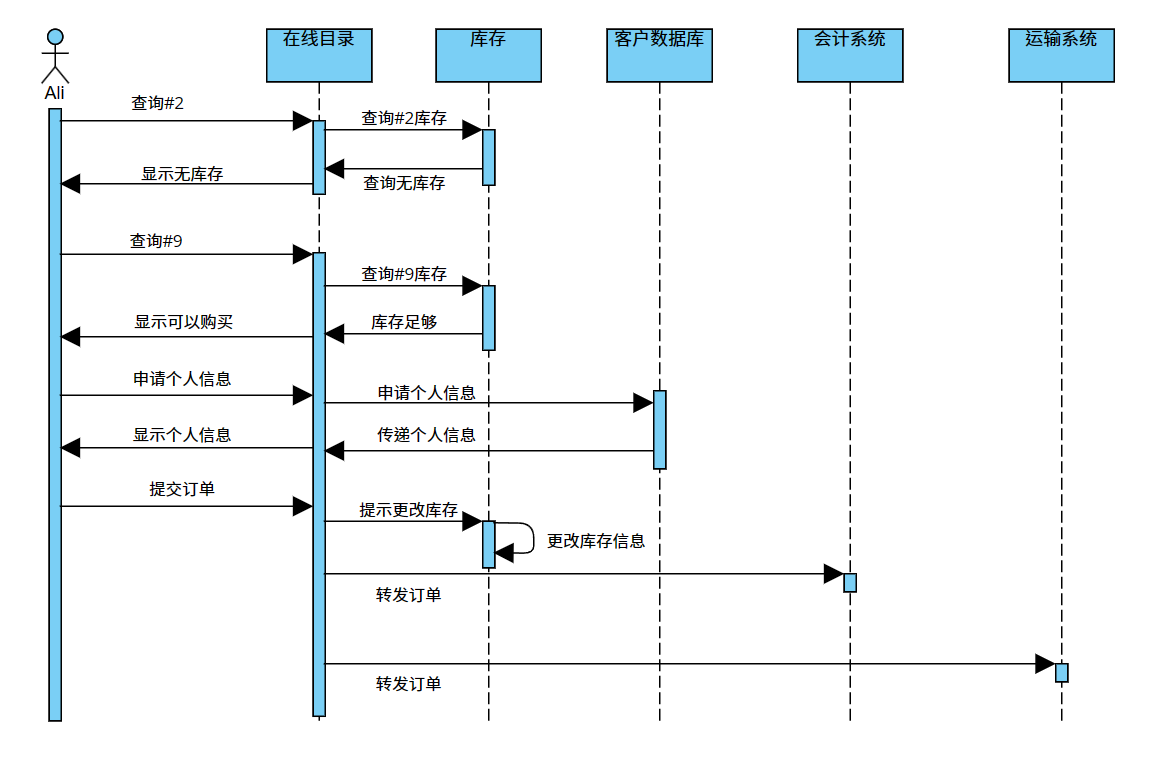
（2）修改前：（花费时间：45min）



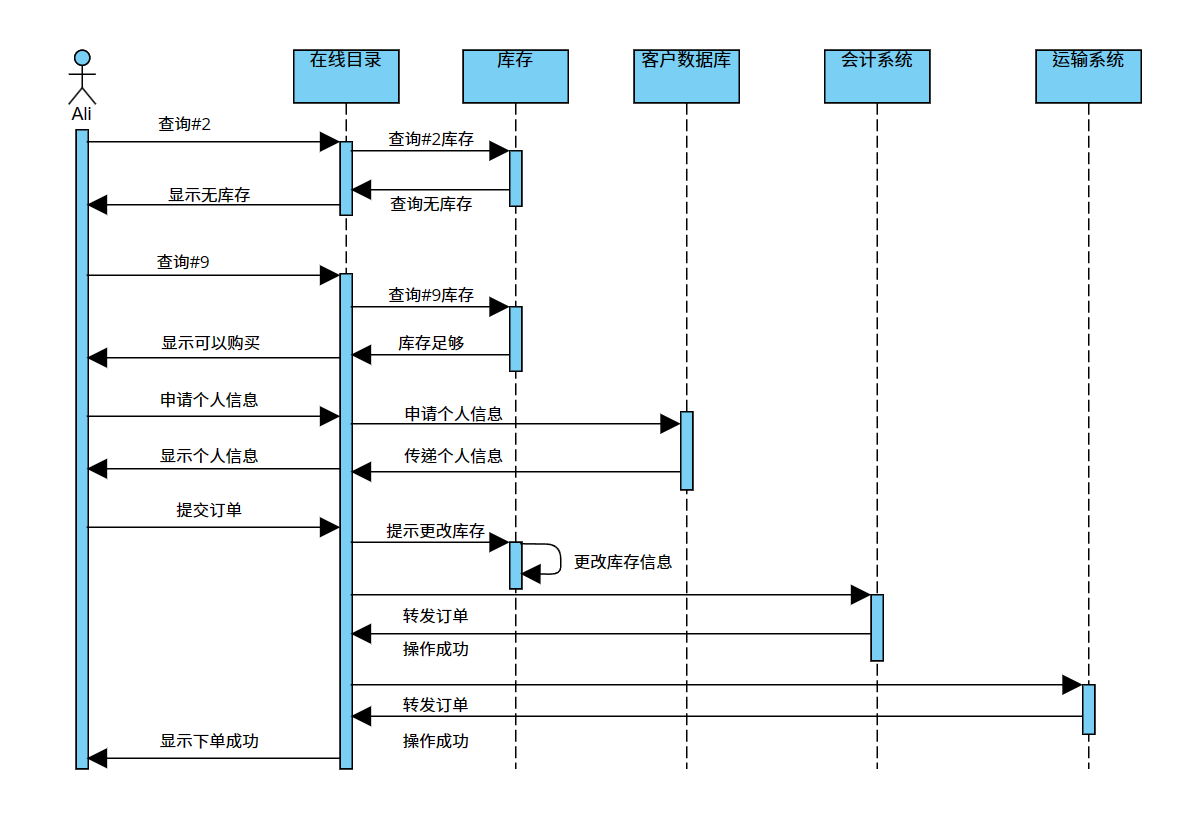
修改后：（花费时间：10min）



（3）修改前：（花费时间：30min）



修改后：（花费时间：10min）



对话过程：

**我：（描述题目），请为上面描述的系统识别参与者和用例。**

Ai：首先，我们需要识别问题描述中涉及的主要参与者（Actor）和用例（Use Case）。

参与者（Actors）

客户（Customer）：通过web或电话与系统进行交互，下订单、检查订单状态、取消订单、请求目录和退货（仅通过电话）。

客户代表（Customer Representative）：接听电话，帮助客户与系统进行交互。

会计系统（Accounting System）：处理订单的成本和发票信息。

运输公司（Shipping Company）：负责订单的填充和运输。

用例（Use Cases）

下订单（Place Order）：客户选择产品，提供送货地址和信用卡信息，然后提交订单。

检查订单状态（Check Order Status）：客户查看订单的当前状态。

取消订单（Cancel Order）：客户取消未处理的订单。

请求目录（Request Catalog）：客户请求产品目录。

退货（Return Product）：客户通过电话退货。

接收订单信息（Receive Order Information）：会计系统接收订单信息并处理发票。

发送订单（Ship Order）：运输公司接收订单并处理配送。

提升客户等级（Upgrade Customer Status）：根据客户的消费金额，系统自动升级客户等级为黄金客户。

注册缺货通知（Register Back-in-Stock Notification）：黄金客户在特定产品缺货时注册接收到货通知。

**我：是否还可以挖掘出更多用例？**

Ai：确实，从问题描述中我们可以进一步挖掘出更多的用例。考虑到系统的自动化和增强功能的需求，以下是一些可能的额外用例：

客户相关用例

管理个人信息：客户可以更新或查看他们的个人信息，如姓名、地址、联系方式等。

查看购买历史：客户可以查看他们过去的订单和购买记录。

积分管理：黄金客户可以查看和管理他们的积分，包括积分获取、使用和余额。

偏好设置：客户可以设置他们的偏好，例如默认的送货地址、支付方式或接收营销邮件的偏好。

订单处理相关用例

重新订购：客户可以快速重新订购以前购买过的商品。

订单跟踪：客户可以实时跟踪他们的订单状态，包括订单是否已发货、预计到达时间等。

批量购买优惠：对于大量购买同一商品的客户，系统可以提供批量购买折扣或优惠。

系统维护与管理用例

用户权限管理：管理员可以管理不同用户的权限，如客户代表、系统管理员等。

订单异常处理：当订单处理中出现问题时（如库存不足、支付失败等），系统管理员或客户代表可以进行异常处理。

产品目录维护：管理员可以更新和维护产品目录，包括添加新产品、更新价格或描述、管理库存等。

扩展功能用例

推荐产品：基于客户的购买历史和偏好，系统可以向客户推荐相关产品或服务。

优惠券和促销活动：系统可以管理优惠券和促销活动，客户可以在购买时使用优惠券或参与促销活动。

客户服务支持：提供客户服务支持渠道，如在线客服、常见问题解答等，以帮助客户解决问题和获取帮助。

这些额外的用例可以根据实际业务需求和系统设计进一步细化和完善。通过不断挖掘和细化用例，可以确保系统能够满足客户的各种需求，并提供更好的用户体验。

**我：快速进行应用领域分析，得出上述系统的对象模型。用UML类图表达您的发现，确保识别类的任何关键操作。**

Ai：

理解该系统的对象模型需要考虑系统的各个组成部分以及它们之间的交互。以下是一个简化的 UML 类图，展示了系统中的主要对象及其关系：

解释一下：

Customer（客户）：代表购买产品的个人或实体。拥有唯一的客户号码、姓名和地址。可以是普通客户或黄金客户。有下订单、检查订单状态、取消订单等操作。

Order（订单）：代表客户的订单。包含订单号、订单日期、订单状态等属性。与 Customer 之间是一对多的关系，即一个客户可以拥有多个订单。具有提交订单、取消订单等操作。

Product（产品）：公司出售的产品。具有产品编号、名称、描述、价格等属性。与 Order 之间是多对多的关系，即一个订单可以包含多个产品，一个产品也可以在多个订单中出现。

Inventory（库存）：存储产品的库存信息。包含产品编号、可用数量等属性。与 Product 是一对一或一对多的关系，取决于是否允许某个产品有多个库存地点。

Catalog（目录）：包含可供客户选择的产品列表。与 Product 是一对多的关系，即一个目录包含多个产品。

ShippingCompany（运输公司）：负责订单的配送和运输。与 Order 是一对多的关系，即一个订单对应一个运输公司。

Payment（支付）：客户用于支付订单的支付信息。包含信用卡号、支付金额等属性。

AccountingSystem（会计系统）：处理订单支付信息的系统。接收订单信息并生成发票。

GoldCustomer（黄金客户）：消费额度超过一定金额的客户。拥有普通客户的所有特征，还有额外的权益，如退货期限延长、积分增加等。

EmailNotification（电子邮件通知）：用于通知黄金客户特定产品到货的通知。与 GoldCustomer 是一对一的关系。

Return（退货）：客户的退货信息。包含退货原因、退货数量等属性。与 Customer 是一对多的关系，即一个客户可以有多个退货记录。

**我：（描述第三问）**

Ai：由于在这里无法直接绘制UML序列图，我将提供一个详细的UML序列图描述，该描述涵盖了Ali下订单的场景。

UML序列图描述

参与者

Ali (客户)

订单处理系统 (OrderProcessingSystem)

客户数据库 (CustomerDatabase)

库存系统 (InventorySystem)

信用卡处理系统 (CreditCardProcessingSystem)

运输系统 (ShippingSystem)

会计系统 (AccountingSystem)

生命线

Ali

OrderProcessingSystem

CustomerDatabase

InventorySystem

CreditCardProcessingSystem

ShippingSystem

AccountingSystem

消息

Ali -> OrderProcessingSystem: 下订单请求 (产品编号#2, 数量1)

OrderProcessingSystem -> InventorySystem: 检查产品可用性 (产品编号#2, 数量1)

InventorySystem -> OrderProcessingSystem: 响应产品不可用

OrderProcessingSystem -> Ali: 显示产品#2不可用消息

Ali -> OrderProcessingSystem: 下订单请求 (产品编号#9, 数量2)

OrderProcessingSystem -> InventorySystem: 检查产品可用性 (产品编号#9, 数量2)

InventorySystem -> OrderProcessingSystem: 响应产品可用

OrderProcessingSystem -> Ali: 显示产品#9可用及详细信息

Ali -> OrderProcessingSystem: 确认订单并提交 (送货地址, 账单地址, 信用卡信息)

OrderProcessingSystem -> CustomerDatabase: 验证客户信息 (送货地址, 账单地址, 信用卡信息)

CustomerDatabase -> OrderProcessingSystem: 验证成功，返回客户详细信息

OrderProcessingSystem -> CreditCardProcessingSystem: 验证信用卡信息 (信用卡号, 账单地址)

CreditCardProcessingSystem -> OrderProcessingSystem: 信用卡验证成功

OrderProcessingSystem -> ShippingSystem: 创建运输订单 (送货地址, 产品信息)

ShippingSystem -> OrderProcessingSystem: 运输订单已创建

OrderProcessingSystem -> AccountingSystem: 发送发票 (信用卡号, 账单地址, 订单成本)

AccountingSystem -> OrderProcessingSystem: 发票已发送

OrderProcessingSystem -> Ali: 显示订单成功消息

备注:

在此序列图中，我们假设所有系统间的通信都是同步的，尽管在实际应用中某些通信可能是异步的。

对于用户身份验证的步骤（如登录过程），由于描述中明确指出可以忽略，所以在序列图中未包含。

订单处理系统还可能会根据Ali的购买历史检查他是否为黄金客户，并据此提供额外的服务或优惠，但这部分在描述中未明确提及，因此也未在序列图中包含。

序列图中的消息可以是调用、返回、创建、发送、接收等不同类型的消息，具体取决于系统设计和实现。